

Chapter 1: Understanding applications

This chapter introduces you to the Meridian IVR application concepts and shows you how to integrate the elements of the application into an effective call flow.

The elements of a Meridian IVR application

To build a voice application with Meridian IVR, you need to use a variety of components that include the following:

- cells
- branches
- buffers
- information databases
- Direct Number Information System (DNIS) support
- Agent Whisper
- identification numbers for prompts and messages

A Meridian IVR application consists of cells that perform a set of interconnected activities during a telephone call. The cell types determine the activities you perform at each step.

You can run an application on one channel or several voice channels at the same time. The application usually begins by answering or making a telephone call, and ends by hanging up the call. What happens in between is entirely up to you as the application developer.

Cells

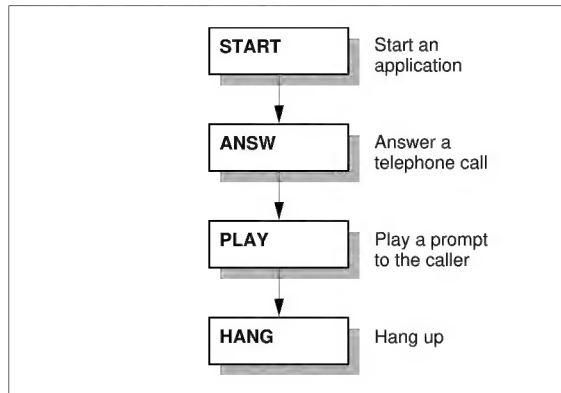
Cells are the building blocks of your application. Each cell performs an action. The cells are linked together by pathways. An application follows these pathways and performs a series of actions.

Meridian IVR cells can perform many different activities within an application. For each kind of activity, there is a cell type. When you are building an application, you choose cell types that perform the activities you want.

An example of an application

The following call-flow diagram, Figure 1-1, represents a simple application that answers a telephone call, plays a prompt to the caller, then hangs up the call.

Figure 1-1
Example of a simple application



In the call flow diagram, each rectangle represents a cell. Inside the rectangle is the cell type such as ANSW for Answer, PLAY for Play Prompts, and HANG for Hang Up. The cell at the top of the call flow is the first cell in the application. The arrows guide you from one cell to the next.

Branches

A branch is a pathway that connects one cell to another. Each branch includes a source and a destination connector (also referred to as an entry connector). A cell can have zero or many source connectors (that is, exception or success connectors), but only one destination connector. You can designate a branch from only a source connector.

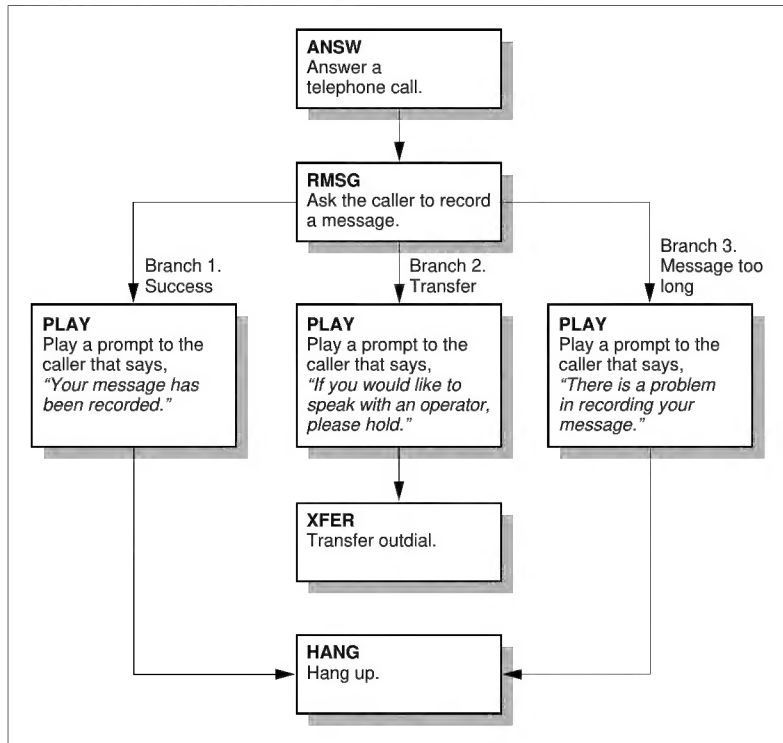
An application must be able to respond to unexpected results. For example, suppose an application asks a caller to leave a message. There is a cell type named Record Message (RMSG) that can do this. However, it is not sufficient to leave it at that; you need to anticipate what a caller can do.

The RMSG cell type has three branches to cover all of the possibilities:

- The caller can successfully leave a message.
- There can be a problem that prevents the system from recording the message.
- The caller can take too long to record the message.

As shown in Figure 1-2, each branch leads to a PLAY cell that plays a prompt appropriate to the situation.

Figure 1-2
An application with branching



Buffers

Meridian IVR uses buffers to store information and pass it from one cell to another. The information stored in a Meridian IVR buffer is a string of up to 31 characters. These characters can be letters, numbers, spaces, or any characters that you type on a keyboard, except for commas.

For example, a cell in an application can ask a caller to enter a password by pressing the keys on the telephone. If the caller presses the 3, 7, 2, and 9 keys, the cell accepts the number 3729 and stores it in a buffer. Now that the password is in a buffer, any other cell can use it. You can add another cell that validates the password.

You can use two different kinds of buffers in your application: system buffers that Meridian IVR creates for you, and user-defined buffers that you create yourself.

System buffers

A set of predefined system buffers is available to each application during each phone call. Each buffer is designed for a special purpose; for example, the buffer called PASSWORD is generally used to store a password that the caller enters. However, you are not restricted to use it for this purpose. Generally, you can use buffers in any way you choose.

The following is a list of system buffers and an explanation of their typical use.

ANI DIGITS

The number of the person who is calling the application.

CURRENT MESSAGE

The identification number of the current message.

DATA EXCHANGE #1

Information that a DELV cell can pass to a scheduled application. You can also use this buffer to store generic data during an application.

DATA EXCHANGE #2

Information that a DELV cell can pass to a scheduled application. You can also use this buffer to store generic data during an application.

DATA EXCHANGE #3

Information that a DELV cell can pass to a scheduled application. You can also use this buffer to store generic data during an application.

DATA EXCHANGE #4

Information that a DELV cell can pass to a scheduled application. You can also use this buffer to store generic data during an application.

DATA EXCHANGE #5

Information that a DELV cell can pass to a scheduled application. You can also use this buffer to store generic data during an application.

DATA EXCHANGE #6

Area where you can store generic data during an application.

DATA EXCHANGE #7

Area where you can store generic data during an application.

DATA EXCHANGE #8

Area where you can store generic data during an application.

DATA EXCHANGE #9

Area where you can store generic data during an application.

DATA EXCHANGE #10

Area where you can store generic data during an application.

DATE

The current date in the form *mmdyyy*.

DAY

The numeric form of the current day of the week, where 1 is Monday and 7 is Sunday.

DELIVERY ATTEMPTS

The number of times a delivery application should try to make the delivery.

DELIVERY EVENT ID

An identification number for a specific delivery event.

DELIVERY INTERVAL

The length of time between delivery attempts.

DELIVERY TIME

The scheduled date and time for delivery.

DIGITS

The number being called.

LENGTH

The length of the string in a buffer.

MAILBOX ID

The identification number of a mailbox which stores messages. The mailbox ID is often a phone number or extension.

NUMBER OF ANI DIGITS

The number of digits in the ANI DIGITS buffer.

NUMBER OF DIGITS

The number of digits in the DIGITS buffer.

NUMBER OF MESSAGES

The number of messages currently stored in a particular mailbox.

PASSWORD

A password associated with a mailbox.

SENDER ID

The identification of the message sender.

TERMINATION DIGIT

The terminating digit from the digit string requested in the GDAT cell.

TIME

The current time in the form of *hhmm*.

Certain system buffers that are automatically updated by Meridian IVR can also be referred to as system-defined buffers.

The following buffers are automatically updated by the corresponding calls. The DIGITS and NUMBER OF DIGITS buffers are updated in the MENU cell. The CURRENT MESSAGE buffer is updated in the RMSG cell and is used by the PMSG and DMSG cells.

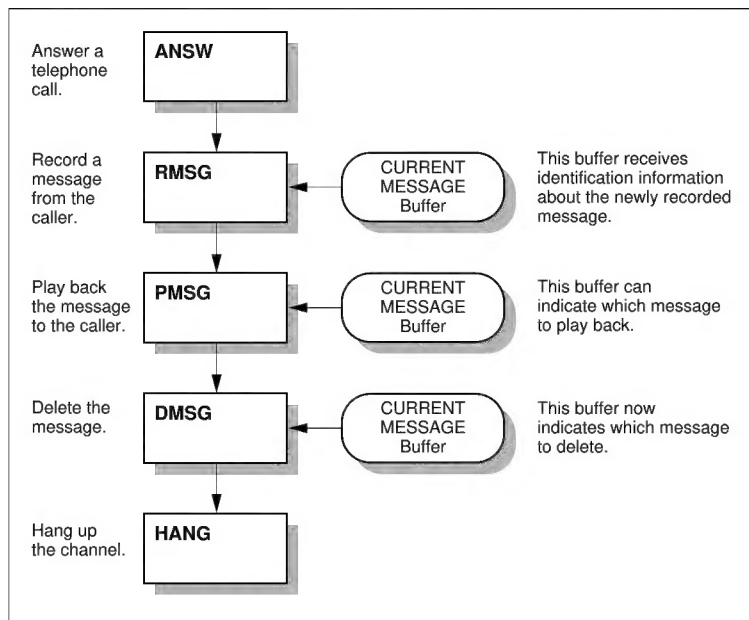
Two of the system buffers deserve special attention: DIGITS and ANI DIGITS. At the beginning of a call, before the application executes the first cell, Meridian Mail can receive digits that represent the called number and the calling number. Meridian IVR automatically takes the called number and stores it in the DIGITS buffer. The number of digits in the DIGITS buffer is automatically stored in the NUMBER OF DIGITS buffer.

Meridian IVR takes the calling number and stores it in the ANI DIGITS buffer. The number of digits in the ANI DIGITS buffer is automatically updated in the NUMBER OF ANI DIGITS buffer. If the Meridian 1 PBX is connected to the telephone network with ISDN trunks, the ANI DIGITS buffer contains Calling Line Identification (CLID) of the caller. For non-ISDN calls, the ANI DIGITS buffer contains the trunk route access code and the trunk route member number. For internal calls, the ANI DIGITS buffer contains the Directory NUMBER (DN) of the caller.

The following call flow, Figure 1-3, shows an application that uses system-defined buffers. This application answers a telephone call, records a message from the caller, plays the message back to the caller, erases the message from Meridian Mail, and hangs up the call.

In this call flow, each oval represents a buffer. An arrow leading from a buffer to a cell indicates that the cell is using the information in the buffer. A black arrow is the pathway from one cell to the next.

Figure 1-3
An application using buffers



User-defined buffers

Meridian IVR allows you to use as many as 50 buffers of your own in any single application. You can create a user-defined buffer at any point in the application. Once it is created, the user-defined buffer is available to all the cells in the application.

There are three reasons why you may want to create user-defined buffers:

- You find that your application needs many buffers, and there are not enough system buffers for your purposes.
- You want to have a buffer whose name reminds you of its specific function in the application, and none of the system buffers seems appropriate.
- You want to have a buffer that is entirely under your control. Some cell types use buffers automatically. However, you can prefer just to create a buffer that you know will not be used unless you explicitly reference it.

If you are not a programmer, you only need to remember that a buffer can store up to 31 characters. These characters can be letters, numbers, spaces, or any of the other characters that you can type on a keyboard except commas.

If you are a programmer, you need to remember that each buffer contains a null-terminated string of up to 31 characters which can be any ASCII character except commas.

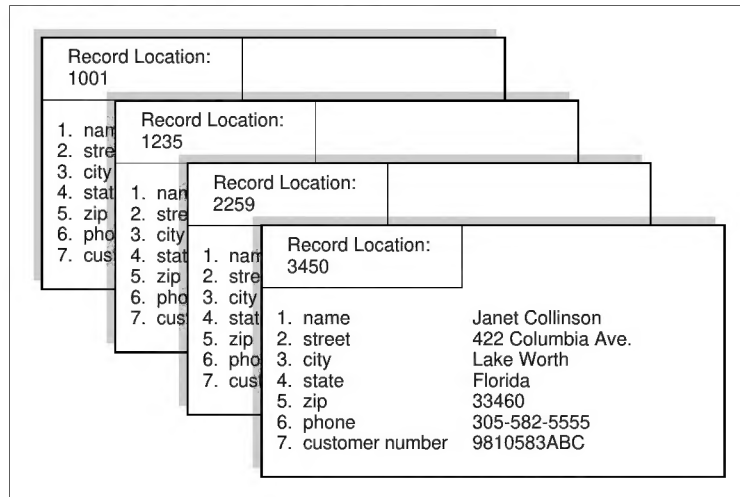
Buffers are created by accessing a cell where a buffer resides and renaming the buffer.

Information databases

An information database consists of a set of records, each of which stores strings of characters. Figure 1-4 illustrates an information database. You can create more than one information database. Up to 20 information databases can be in use by all of the applications that are running at any one time.

To use the information, an application can have a Check database cell (CHEK) to copy data from one record to a set of buffers you have chosen. Once the data has been placed in buffers, the application can use the data just as it would any other buffer data. For example, the application could look up a given record in a database, copy a phone number from the record to a buffer in the application, and then read the phone number to the caller.

Figure 1-4
Records in an information database



Directory Number Identification System

Directory Number Identification System (DNIS) identifies to the called system the last three or four digits of the number dialed by the caller.

One example of the use of DNIS is with 800 numbers. A company has set up several 800 numbers, one for testing your advertisements on TV stations in Phoenix, another for testing your advertisements on TV stations in Chicago, and another for Milwaukee. A Local Telephone Carrier terminates all the lines in one group to your IVR ACD because it is cheaper and more efficient to run one group of incoming lines. The calls are all answered by IVR applications, and the DNIS for each call directs the call to the appropriate application, which is invoked as a subroutine within an application.

The DNIS is stored in the system buffers of an application. These system buffers are DIGITS and ANI. At the beginning of a call, before the first cell in the application is executed, Meridian Mail can receive DNIS information regarding the called and calling numbers. Meridian IVR automatically takes and stores the called number information in the DIGITS buffer and stores the calling number information in the ANI buffer.

Agent Whisper

Agent Whisper allows the system to play a prerecorded phrase and/or convey call-related information to an agent.

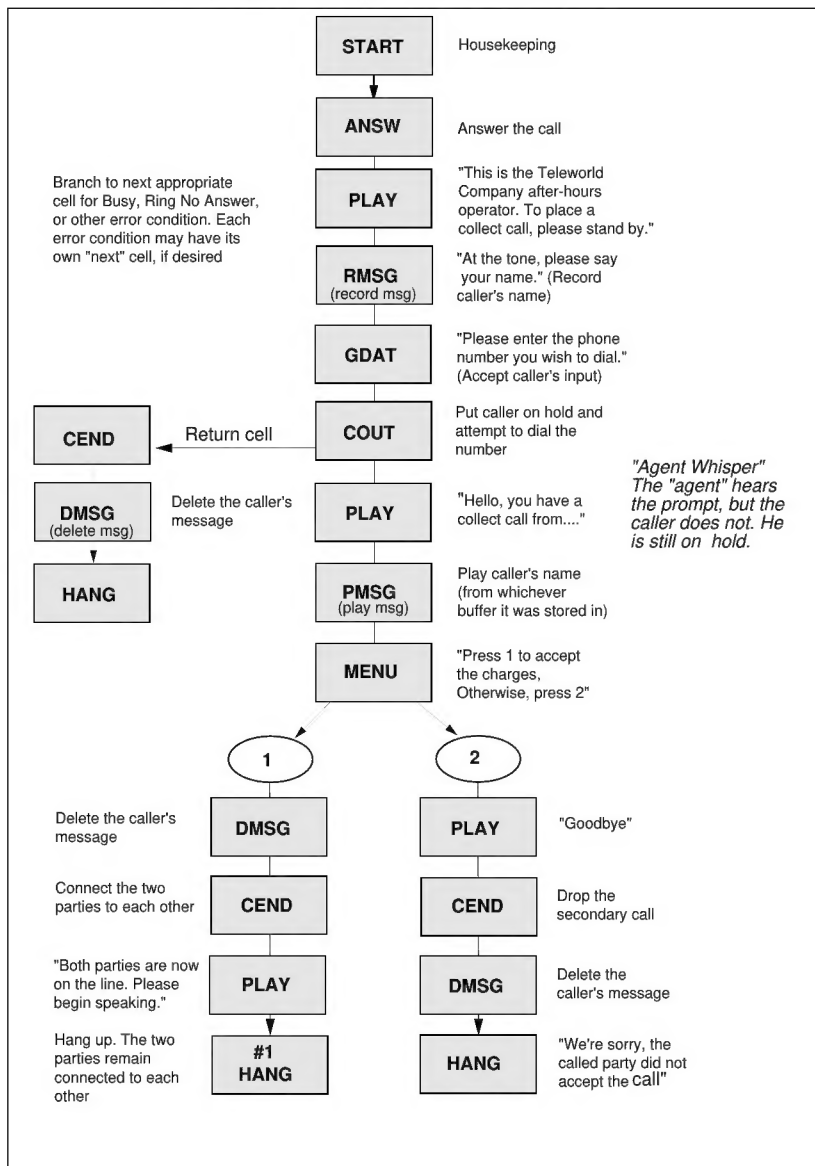
Agent Whisper is most commonly used in collect calling. A person placing a collect call is prompted to say their name into the phone where it is recorded. Then the call is processed and rings the designated number. When the phone is answered, a recorded voice informs whoever answers the phone that there is a collect call from “Name”, and then asks them to accept the collect charge. When the computer plays back the recording of the caller’s name, the caller does not hear this. This is why it is called Agent Whisper.

Agent Whisper is also used to relay information about a call to a call attendant or agent. This could be information about a series of menu choices that a caller has selected so that the Agent is prepared for the call when it is transferred. The information that is conveyed is application dependent. This information can be the DNIS number, the nature of the call, or the length of time that a caller has been waiting.

Prompts and information are relayed to the agent using a PLAY cell between the Continuation Outdial (COUT) and Continuation Outdial End (CEND) cells. Only the agent hears what the PLAY cell plays, not the caller.

Figure 1-5 shows an application that uses Agent Whisper.

Figure 1-5
Using Agent Whisper in an application



Identification numbers for prompts and messages

All voice applications must manage two kinds of voice recordings: prompts and messages. Prompts are played to callers to guide them through applications created in advance using the Voice Prompt Editor (VPE). Messages are created by callers and typically have a short life-span.

When a message is recorded, Meridian IVR automatically assigns a number to the message. You never need to know the number of a message because Meridian IVR takes care of message numbers automatically.

When you want to use prompts, you need to know what prompt numbers to use. Refer to the VPE to view the available prompt numbers.